Fast Exact Stochastic Simulation Algorithms Using Partial Propensities

Ramaswamy, Rajesh ; Sbalzarini, Ivo F

Abstract: We review the class of partial-propensity exact stochastic simulation algorithms (SSA) for chemical reaction networks. We show which modules partial-propensity SSAs are composed of and how partial-propensity variants of known SSAs can be constructed by adjusting the sampling strategy used. We demonstrate this on the example of two instances, namely the partial-propensity variant of Gillespie's original direct method and that of the SSA with composition-rejection sampling (SSA-CR). Partial-propensity methods may outperform the corresponding classical SSA, particularly on strongly coupled reaction networks. Changing the different modules of partial-propensity SSAs provides flexibility in tuning them to perform particularly well on certain classes of reaction networks. The framework presented here defines the design space of such adaptations.

DOI: https://doi.org/10.1063/1.3497968

Fast Exact Stochastic Simulation Algorithms Using Partial Propensities

Rajesh Ramaswamy and Ivo F. Sbalzarini

Citation: AIP Conference Proceedings **1281**, 1338 (2010); doi: 10.1063/1.3497968
View online: http://dx.doi.org/10.1063/1.3497968
View Table of Contents: http://scitation.aip.org/content/aip/proceeding/aipcp/1281?ver=pdfcov
Published by the AIP Publishing

Fast Exact Stochastic Simulation Algorithms Using Partial Propensities

Rajesh Ramaswamy and Ivo F. Sbalzarini

Institute of Theoretical Computer Science and Swiss Institute of Bioinformatics,
ETH Zürich, Universitätstr. 6, CH-8092 Zurich, Switzerland

Abstract. We review the class of partial-propensity exact stochastic simulation algorithms (SSA) for chemical reaction networks. We show which modules partial-propensity SSAs are composed of and how partial-propensity variants of known SSAs can be constructed by adjusting the sampling strategy used. We demonstrate this on the example of two instances, namely the partial-propensity variant of Gillespie's original direct method and that of the SSA with composition-rejection sampling (SSA-CR). Partial-propensity methods may outperform the corresponding classical SSA, particularly on strongly coupled reaction networks. Changing the different modules of partial-propensity SSAs provides flexibility in tuning them to perform particularly well on certain classes of reaction networks. The framework presented here defines the design space of such adaptations.

Keywords: stochastic simulation algorithm, SSA, partial propensities, chemical reactions, partial propensity methods
PACS: 82.20.-w, 87.10.Rt, 87.10.Mn

INTRODUCTION

Stochastic chemical kinetics are typically described by the chemical master equation (CME). Numerical simulations of the CME can be done using the stochastic simulation algorithm (SSA) [1]. SSA is governed by the joint probability density for the time to the next reaction and the index of the next reaction.

Any chemical reaction network with N species and M reactions can be represented by its dependency graph. Each node in this graph represents a chemical reaction and a directed edge is drawn from node p to node q if firing of reaction p affects the copy number of any of the reactants of reaction q. In this representation, we quantify the degree of coupling of the reaction network as the maximum number of edges that leave any node, i.e., the maximum out-degree of the dependency graph. Using this quantification, we classify chemical reaction networks into *weakly* and *strongly* coupled networks [2, 3]. Weakly coupled networks have a degree of coupling that is bounded by a constant with increasing network size. Strongly coupled networks have a degree of coupling that increases unboundedly with network size. The computational cost of exact SSA formulations depends on the coupling class of the reaction network. For weakly coupled reaction networks the computational cost (CPU time) has been reduced to $O(1)$ [3, 4] under the assumption that the ratio of maximum to minimum propensity is bounded by a constant. For strongly coupled networks, the computational cost has been reduced from $O(M)$ to $O(N)$ using the concept of partial propensities as introduced by Ramaswamy et al. [2] and later independently also by Indurkhya et al. [5].

Here, we review the class of partial propensity methods and present their modular building blocks. We show that by modifying certain modules one can flexibly obtain different partial propensity formulations, each of which being particularly efficient on a certain class of reaction networks. For example, on weakly coupled reaction networks, the partial propensity SSA with composition-rejection sampling (PSSA-CR) has a computational cost of $O(1)$ under the assumption that the ratio of maximum to minimum propensity is bounded by a constant. On strongly coupled reaction networks, the partial propensity direct method (PDM) is particularly efficient with a computational cost of $O(N)$. On multi-scale networks, the sorting variant of PDM (SPDM) is recommended.

MODULES OF PARTIAL-PROPENSITY ALGORITHMS

Partial propensity SSAs are based on the concept of factored-out, partial reaction propensities. These are defined as the propensity of a reaction per unit number of molecules of one of its reactants [2, 3]. Bimolecular reactions thus have two partial propensities, unimolecular reactions one. Using partial propensities to sample the next reaction amounts

CP1281, *ICNAAM, Numerical Analysis and Applied Mathematics, International Conference 2010, Vol.II*
edited by T. E. Simos, G. Psihoyios, and Ch. Tsitouras

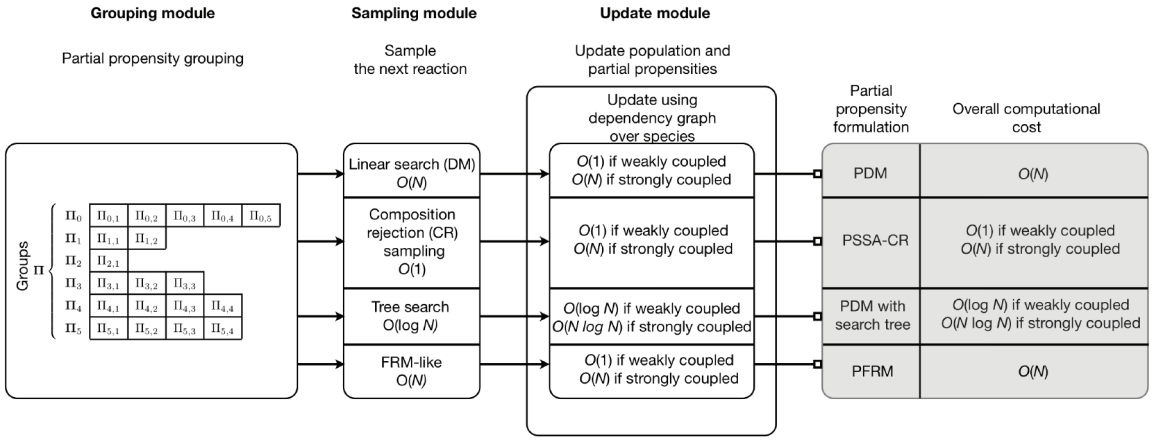

FIGURE 1. Illustration of the three modules of partial propensity SSAs: the grouping module, the sampling module, and the update module. The grouping module groups reactions according to their partial propensities. The sampling module samples the next reaction. The update module updates the population and the affected partial propensities after a reaction has fired. Using different algorithms in the sampling module we obtain different partial propensity formulations, such as PDM (the partial-propensity direct method), PSSA-CR (the partial-propensity SSA with composition-rejection sampling), PDM with search trees, or PFRM (the partial-propensity first reaction method). These formulations have varying overall computational costs, depending on the coupling class of the reaction network. Note: N is the number of chemical species in a reaction network.

to sampling *reaction partners* instead of complete reactions. In a network that consists of M reactions, only $N < M$ distinct reaction *partners* exist. The partial propensities of the three elementary reaction types are:

- Bimolecular reactions ($S_i + S_j \rightarrow$ Products): $a_\mu = n_i n_j c_\mu$ and $\pi_\mu^{(i)} = n_j c_\mu$, $\pi_\mu^{(j)} = n_i c_\mu$.
  If both reactants are of the same species, i.e. $S_i = S_j$, only one partial propensity exists, $\pi_\mu^{(i)} = \frac{1}{2}(n_i - 1)c_\mu$.
- Unimolecular reactions ($S_i \rightarrow$ Products): $a_\mu = n_i c_\mu$ and $\pi_\mu^{(i)} = c_\mu$.
- Source reactions ($\emptyset \rightarrow$ Products): $a_\mu = c_\mu$ and $\pi_\mu^{(0)} = c_\mu$.

The use of partial propensities can be interpreted as follows: Let $\mathbf{D}$ be the $N \times N$ diagonal matrix of species populations with element $D_{i,i}$ the population of species i. Further, let $\mathbf{B}$ be the symmetric, positive definite $N \times N$ matrix of specific probability rates of all bimolecular reactions. Element $B_{i,j} = B_{j,i} > 0$ is the specific probability rate c of the reaction of species i with species j. Similarly, the specific probability rates of all unimolecular reactions are collected in the $N \times N$ diagonal matrix $\mathbf{U}$. The propensities of bimolecular reactions are then given by the product $\mathbf{A_B} = \mathbf{DBD}$, those of unimolecular reaction by $\mathbf{A_U} = \mathbf{UD}$. Traditional SSA formulations amount to first explicitly computing all propensities and then sampling on $[\mathbf{A_B}, \mathbf{A_U}]$. Partial propensity methods sample on the partial product $\mathbf{BD}$ and multiply the sampling result with the corresponding $D_{i,i}$ *afterward*. This is implemented using three modules: grouping the partial propensities, sampling the next reaction, and updating the values. These modules of partial propensity SSAs are summarized in Fig. 1. Different partial-propensity methods with different computational costs can be constructed by using different algorithms in the sampling module. For a detailed cost analysis of PDM and PSSA-CR we refer to the original publications [2, 3].

Grouping module. Partial propensity methods group the partial propensities of all reactions according to the index of the factored-out reactant, i.e., the common reaction partner. Each group thus contains the partial propensities of all reactions having this species as a reactant. The different partial propensities within a group correspond to the various possible reaction partners of the common, factored-out reactant. For any reaction network, there are at most $N+1$ groups (including group 0 for source reactions) and the number of partial propensities in each group is at most $O(N)$.

For higher-order reactions (trimolecular and more), multi-dimensional grouping can be used with one dimension per reactant. Again, the total number of groups in each dimension is $O(N)$ and the sampling module is independently applied along each direction in order to sample the reaction partners.

Sampling module. The key building block of partial propensity methods is the algorithm used to sample the next reaction. Given the grouping of partial propensities, this involves sampling the index of the group and then the index of the element within the group. Sampling the index of the group amounts to sampling the first reactant of the next reaction. In order to find out which partner this reactant will react with, the partial propensity within the group is sampled. For unimolecular and source reactions, the partial propensities are constants and the second step is omitted.

All sampling algorithms used in standard SSAs can also be used in partial-propensity methods. Instead of applying them over reactions, however, they are first applied over partial-propensity groups and then over the elements within the selected group. For example, using linear search (as in Gillespie's direct method [1]) leads to a sampling step that is $O(N)$ on all classes of networks. Replacing linear search by composition-rejection sampling [6] reduces the computational cost of the sampling step to $O(1)$. Other sampling strategies, such as search trees or a first-reaction-method-like sampling over the reaction times can also be used straightforwardly. Depending on the sampling strategy and the associated algorithmic overhead, certain partial-propensity formulations are particularly well suited for certain classes of reaction networks (Fig. 1).

Update module. After the selected reaction has fired and the populations of the involved species have been updated, the affected partial propensities are recomputed using a dependency graph over species. Since any partial propensity is a function of the population of at most one species, the number of partial propensities to be updated is at most $O(N)$. In weakly coupled reaction networks, the number of partial propensities to be updated is $O(1)$, since the degree of coupling is bounded by a constant. However, depending on the data structures that are used in the sampling module, the computational cost of the update module varies. Figure 1 shows the computational cost of the update step depending on the sampling method used.

BENCHMARKS

We demonstrate the computational performance of PDM and PSSA-CR on both a weakly coupled and a strongly coupled reaction network. We choose the cyclic chain and the colloidal aggregation model as representative networks, respectively, as previously described [2, 3].

The cyclic chain is the most weakly coupled network possible. For N chemical species, it has the minimum possible number of $M = N$ reactions. The degree of coupling is 2, independently of the length of the chain (i.e., the system size N). The colloidal aggregation model is an example of a network with half-full matrix $\mathbf{B}$. For N chemical species, the number reactions in this network is $M = \left\lfloor \frac{N^2}{2} \right\rfloor$. The degree of coupling of this reaction network is $3N-7$ and hence scales with system size (strong coupling).

For all test cases, we simulate the reaction network until 10^7 reactions have been executed and report the average CPU time Θ per reaction. At time $t = 0$, we set all $n_i = 1$ and all specific probability rates $c_i = 1$. Fig. 2A shows the results for SPDM and PSSA-CR on the weakly coupled cyclic chain network. As expected from the sampling algorithm used, Θ is $O(1)$ for PSSA-CR and $O(N)$ for SPDM in this case. For the strongly coupled colloidal aggregation network, Θ is $O(N)$ for both SPDM and PSSA-CR. The absolute Θ of PSSA-CR, however, is always larger than that for SPDM, due to the additional algorithmic overhead of composition-rejection sampling. By modularly combining the building blocks of partial propensity methods, they can thus be tuned to specific classes of reaction networks.

This flexibility can lead to significant computational savings in practical applications. Even for networks of moderate size, partial propensity formulations can be computationally more efficient than the corresponding classical SSA formulations [2, 3]. For large reaction network, such as metabolic networks or protein interaction networks, the speedup will be more and more significant. For example, 100 Monte Carlo runs of 10 million reactions of the strongly-coupled protein interaction network of yeast (4000 species and 16000 reactions) or drosophila (7000 species and 20000 reactions) [7] can be simulated in a day using PDM [2], whereas even the most efficient classical SSA for strongly coupled networks [8] would take several months.

PARTIAL PROPENSITIES AND APPROXIMATE SSA

If the number of molecules is sufficiently large, the exponentially weighted time to the next reaction in the exact SSA can be approximated by a Poisson distribution. This enables constructing approximate SSAs that execute several reactions per iteration and are hence computationally more efficient. This comes at the price of loosing

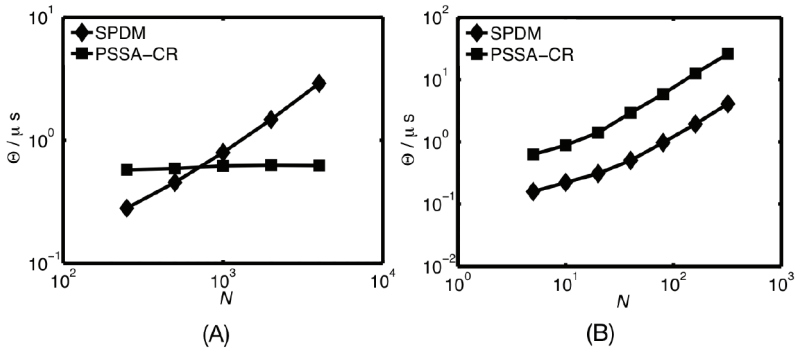

FIGURE 2. Computational cost of SPDM (diamonds) and PSSA-CR (squares). The average CPU time Θ per reaction, averaged over 100 independent runs, is shown as a function of the network size N. (A) Θ for the weakly coupled cyclic chain network is $O(1)$ for PSSA-CR and $O(N)$ for SPDM. (B) Θ for the strongly coupled colloidal aggregation model is $O(N)$ for both PSSA-CR and SPDM.

accuracy, especially about higher-order moments of the CME. Higher-order moments are, however, important to understand deviations of the stochastic from the deterministic behavior or to compare to non-equilibrium kinetics from spectroscopy experiments. Since approximate SSAs sample the number of firings for each reaction at every time step their computational cost is $O(M)$ on all classes of networks, albeit with a smaller pre-factor than that of exact SSAs. Using partial propensities would, in our opinion, not change this. We therefore believe that partial-propensity approximate SSAs are only of academic interest.

CONCLUSIONS

We have presented the modules of partial-propensity SSA formulations and have shown how changing the sampling algorithm leads to different partial propensity methods. Since all sampling methods known from classical SSAs can also be used in partial-propensity methods, each known SSA has an associated partial-propensity variant with potentially reduced computational cost. Each of these variants is particularly well suited for chemical reaction networks with certain properties. This provides a modular toolbox for efficient exact stochastic simulations of chemical reaction networks. We have shown how the previously presented partial-propensity direct method (PDM) and the partial-propensity SSA with composition-rejection sampling (PSSA-CR) can be interpreted in the design space of this framework. The computational costs of the various formulations follow straightforwardly from the corresponding modules, as demonstrated in the presented benchmarks. The present framework allows the systematic construction of further partial propensity methods by changing the algorithms used inside each module. This defines the family of partial-propensity methods for exact stochastic simulation of chemical reaction networks.

ACKNOWLEDGMENTS

This project was supported with a grant from the Swiss SystemsX.ch initiative, evaluated by the SNSF.

REFERENCES

1. D. T. Gillespie, *J. Comput. Phys.* **22**, 403–434 (1976).
2. R. Ramaswamy, N. González-Segredo, and I. F. Sbalzarini, *J. Chem. Phys.* **130**, 244104 (2009).
3. R. Ramaswamy, and I. F. Sbalzarini, *J. Chem. Phys.* **132**, 044102 (2010).
4. A. Slepoy, A. P. Thompson, and S. J. Plimpton, *J. Chem. Phys.* **128**, 205101 (2008).
5. S. Indurkhya, and J. Beal, *Plos One* **5**, e8125 (2010).
6. L. Devroye, *Non-uniform random variate generation*, Springer-Verlag New York, 1986.
7. D. Li, J. Li, S. Ouyang, J. Wang, S. Wu, P. Wan, Y. Zhu, X. Xu, and F. He, *Proteomics* **6**, 456–461 (2006).
8. J. M. McCollum, G. D. Peterson, C. D. Cox, M. L. Simpson, and N. F. Samatova, *Comput. Biol. Chem.* **30**, 39–49 (2006).